



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

09/932,541

08/17/2001

John Lacombe

1662-30500 JMH
(P00-2947)

3871

22879

7590

04/12/2005

EXAMINER

COURTENAY III, ST JOHN

HEWLETT PACKARD COMPANY
P O BOX 272400, 3404 E. HARMONY ROAD
INTELLECTUAL PROPERTY ADMINISTRATION
FORT COLLINS, CO 80527-2400

ART UNIT

PAPER NUMBER

2194

DATE MAILED: 04/12/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/932,541

Applicant(s)

LACOMBE ET AL.

Examiner

St. John Courtenay III

Art Unit

2194

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 28 December 2004.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-22 and 27-31 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☒ Claim(s) 9-12 is/are allowed.
- 6) ☒ Claim(s) 1-8, 13-20, 22 and 27-31 is/are rejected.
- 7) ☒ Claim(s) 21 is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 17 August 2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.


ST. JOHN COURTENAY III
PRIMARY EXAMINER

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: _____.

Response to Amendment

Applicant's amendments and supporting arguments have been considered, but are deemed to be moot in view of the new grounds of rejection set forth below.

Objection to the Specification [see MPEP 2001.06(b)]

The individuals covered by **37 CFR 1.56** have a duty to bring to the attention of the examiner, or other Office official involved with the examination of a particular application, information within their knowledge as to other copending United States applications which are "material to patentability" of the application in question. As set forth by the court in *Armour & Co. v. Swift & Co.*, 466 F.2d 767, 779, 175 USPQ 70, 79 (7th Cir. 1972):

See also **MPEP § 2004**, paragraph 9.

Accordingly, the individuals covered by **37 CFR 1.56** cannot assume that the examiner of a particular application is necessarily aware of other applications which are "material to patentability" of the application in question, but must instead bring such other applications to the attention of the examiner. See *Dayco Prod., Inc. v. Total Containment, Inc.*, 329 F.3d 1358, 1365-69, 66 USPQ2d 1801, 1806-08 (Fed. Cir. 2003).

For example, if a particular inventor has different applications pending in which similar subject matter but patentably indistinct claims are present that fact must be disclosed to the examiner of each of the involved applications. Similarly, the prior art references from one application must be made of record in another subsequent application if such prior art references are "material to patentability" of the subsequent application. See *Dayco Prod.*, 329 F.3d at 1369, 66 USPQ2d at 1808.

Appropriation correction is required with respect to copending related application **Lacombe et al.** (U.S. Patent Application Publication US 2002/0184482), filed May 31, 2001.

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claims 1- 8, 13-20, 22, 27-31 are rejected under 35 U.S.C. § 102(e) as being clearly anticipated by **Lacombe et al.** (U.S. Patent Application Publication US 2002/0184482).

Note: Although there is one common inventor with the instant application (inventor John Lacombe), the cited Patent Application Publication (US 2002/0184482) constitutes a different inventive entity than the instant application.

As per independent claim 1:

Lacombe teaches a computer system, comprising at least one processor, a system memory coupled to said processor [§0026], at least one input/output device [§0023, see mouse and keyboard] coupled to said processor, and a watchdog timer device [§0034, see watchdog driver 30], wherein the computer system executes:

- an operating system with at least two protection layers [§0029, e.g., "The NT environment provides two software protection layers: Ring 0 and Ring 3"];
- one or more key computer applications [§00036, "application 440"; and
- an application watchdog driver that monitors user designated computer applications for periodic messages wherein if the watchdog driver receives a periodic message from all user-designated computer applications in a predetermined period of time, the watchdog driver delivers a command to clear the watchdog timer device [§0038, "watchdog driver 360"].

As per independent claim 7:

This claim is rejected for the same reasons detailed above in the rejection of independent claim 1, and also for the following additional reasons:

Lacombe teaches a dedicated watchdog counter in the hardware layer of a computer system, and a watchdog driver operating in the kernel mode of the computer operating system, the watchdog driver comprising:

- a system thread configured to monitor a plurality of designated user applications operating in the user mode of the computer operating system [§0011, "The driver includes a system thread configured to monitor a plurality of user applications that operate in the user mode of the computer operating system"];

- a message passing interface for receiving periodic signals from each of the user applications [§0013, "In the preferred embodiment, the messages from the applications are sent periodically by the applications and directed specifically to the watchdog driver. The messages are preferably sent to the watchdog driver via a message passing interface between the user mode and kernel mode. The message passing interface is preferably implemented as shared memory queues"]; and
- a communication interface for transmitting a timer reset command to the dedicated watchdog counter [§0011, "Lastly, a communication interface is provided for coordinating timer events with the operating system scheduler."];
- wherein if the system thread receives a message from each of the designated user applications within an allotted period of time, the watchdog driver sends a timer reset command to the dedicated watchdog counter and wherein if the system thread does not receive a message from each of the designated user applications within the allotted period of time, the watchdog driver does not send a timer reset command to the dedicated watchdog counter [see restart and system thread discussion §0012].

As per independent claim 16:

This claim is rejected for the same reasons detailed above in the rejection of the preceding independent claims, and also for the following additional reasons:

Lacombe teaches a method of detecting and restarting an unresponsive computer application, comprising:

- executing the application in a first protective layer of a computer operating system [§0029, "Applications running in Ring 3 cannot physically access memory space in the more highly protected Ring 0 layer."];
- executing an application watchdog driver in a second, more protected, protective layer of the computer operating system [see HAL and Ring 0 discussion §0030];
- establishing a message passing interface between the application and the watchdog driver [§0034, "the watchdog driver 360 establishes an initial IOCTL interface 390 that establishes the appropriate message passing interface 350 and a run-time IOCTL signal interface 395 for communication with the application restart service."];
- periodically transmitting signals from the application to the message passing interface [§0013, "In the preferred embodiment, the messages from the applications are sent periodically by the applications and directed specifically to the watchdog driver. The messages are preferably sent to the watchdog driver via a message passing interface between the user mode and kernel mode. The message passing interface is preferably implemented as shared memory queues"];
- executing a system thread in the watchdog driver that is configured to monitor the message passing interface for the periodic signals from said application or other designated applications; and using a dedicated watchdog timer device to count from a programmable initial value to a final system reset value [see restart and system thread discussion §0012];

- wherein if the system thread detects a periodic signal from the application before the watchdog timer counts to the final system reset value, the watchdog driver initiates a command to the watchdog timer to reset the watchdog timer to the initial value and wherein if the system thread fails to detect a periodic signal from the application before the watchdog timer counts to the final system reset value, the watchdog timer initiates a command to restart the computer system [§0036, "During runtime operation the user application sends messages periodically through the interface 350. The watchdog driver system thread 370 will asynchronously monitor the interface 350 for periodic messages from the application 440. If the watchdog driver 360 does not detect a message from the application 330 for a predetermined period of time, the driver 360 will signal the restart service 380 to terminate and restart the application 450."].

As per independent claim 27:

This claim is rejected for the same reasons detailed above in the rejection of the preceding independent claims, and also for the following additional reasons:

Lacombe teaches a computer server, comprising:

- a central processing unit ("CPU") [see CPU 202, §0026] configured to execute an operating system and key designated user applications [§0029, e.g., "The NT environment provides two software protection layers: Ring 0 and Ring 3"; see also application level discussion set forth in §0029; see also §0035, i.e., "Once the restart service 380 is established, the key user application 330 is started and initialized 430. Once the application is linked to an

appropriate DLL 340, the application will call into the DLL 340, which in turn, will make the initialization IOCTL calls 390 into the watchdog driver 360 to establish a connection through the message passing interface or shared memory queues 350."];

- a system memory coupled to said CPU [§0026, "memory 204"];
- an input/output processor ("IOP") configured to control server management architecture [§0034, "the watchdog driver 360 establishes an initial IOCTL interface 390 that establishes the appropriate message passing interface 350 and a run-time IOCTL signal interface 395 for communication with the application restart service."];
- a system watchdog device configured to receive periodic messages from the operating system [§0036, "During runtime operation the user application sends messages periodically through the interface 350. The watchdog driver system thread 370 will asynchronously monitor the interface 350 for periodic messages from the application 440."]; and
- an application watchdog device configured to receive periodic messages from the user applications, wherein if either the system watchdog device or the application watchdog device does not receive a periodic message for a designated period of time, the watchdog device that does not receive the periodic messages initiates a command to the CPU to reset the server [§0036, "If the watchdog driver 360 does not detect a message from the application 330 for a predetermined period of time, the driver 360 will signal the restart service 380 to terminate and restart the application 450."].

As per dependent claim 2:

Lacombe teaches a message passing interface that transmits signals between the two protection layers; wherein the watchdog driver executes in one protection layer and the application executes in another protection layer and wherein the periodic message is transmitted from the application to the application watchdog driver through the message passing interface [§29, "Any communication between applications running in Ring 3 and services in Ring 0 must use a message passing service. This design prevents user applications from interfering with the core NT operating system."].

As per dependent claim 3:

Lacombe teaches the message passing interface is a shared memory queue [§0035, "Once the restart service 380 is established, the key user application 330 is started and initialized 430. Once the application is linked to an appropriate DLL 340, the application will call into the DLL 340, which in turn, will make the initialization IOCTL calls 390 into the watchdog driver 360 to establish a connection through the message passing interface or shared memory queues 350"].

As per dependent claim 4:

Lacombe teaches the watchdog timer device resides in a hardware layer separate from the operating system protection layers and wherein the application watchdog driver communicates with the watchdog timer device via a hardware abstraction layer [§0030, "Also shown in FIG. 3 is a Hardware layer, which represents the physical computer system hardware such as the CPU, timer devices, and watchdog devices" ... "Also included in FIG. 3 is a Hardware Abstraction Layer (HAL) 310, which is used to prevent hardware dependence and provide an isolation layer between the hardware and software. The HAL operates at the Ring 0 level and translates low-level operating system functions into instructions understandable by the physical system

hardware"].

As per dependent claim 5:

Lacombe teaches a system watchdog timer device wherein the computer system also executes a system watchdog driver that monitors the operating system for periodic messages, and wherein if the system watchdog driver receives a periodic message from the operating system in a predetermined period of time, the system watchdog driver delivers a command to clear the system watchdog timer device [§0038, see restarting or reset discussion].

As per dependent claim 6:

Lacombe teaches the watchdog timer devices issue a reset command if either of the watchdog timer devices do not receive a clear timer command from the watchdog drivers in a predetermined period of time [§0038, see restarting or reset discussion].

As per dependent claim 8:

Lacombe teaches, if the watchdog counter does receive a timer reset command from the watchdog driver, the counter is reset to begin counting down from the maximum allotted period of time and wherein if the watchdog counter does not receive a timer reset command from the watchdog driver, the counter is configured to restart the computer system when the counter expires [see reset discussion §§0036-0039].

As per dependent claim 13:

Lacombe teaches the messages from the designated user applications are sent periodically by the applications and directed specifically to the watchdog driver [§0013, "In the preferred embodiment, the messages from the applications are sent periodically by the applications and directed specifically to the watchdog driver. The messages are preferably sent to the

watchdog driver via a message passing interface between the user mode and kernel mode. The message passing interface is preferably implemented as shared memory queues."].

As per dependent claim 14:

Lacombe teaches the plurality of the user applications are prioritized by a computer user to permit varying levels of watchdog protection [§0040, "For example, since the watchdog driver 360 is capable of monitoring several applications, the watchdog system may be configured to provide a user interface to establish priority among the applications."].

As per dependent claim 15:

Lacombe teaches the application watchdog operates in conjunction with a system watchdog that is configured to monitor the computer operating system for periodic activity; and wherein both the application watchdog and the system watchdog are sufficiently configured to restart the computer system if either watchdog does not receive a timer reset command within an allotted period of time [§0039, "the periodic signals sent by the application will be initiated by commands embedded in the computer application software. These commands will be directed at the shared memory queues 350 for the purpose of resetting the application watchdog timer events"].

As per dependent claim 17:

Lacombe teaches sending an early warning message to notify system management software or firmware that the watchdog timer is about to expire [see §0038, timer discussion].

As per dependent claim 18:

Lacombe teaches the initialization of the watchdog driver comprises:

- loading the watchdog driver as the operating system loads following a computer system boot [§0035, "During OS initialization 410, the kernel mode watchdog driver 360 will load and create an initial IOCTL 390 interface with commands for establishing the message passing interface. The watchdog driver 360 will also establish an IOCTL signal interface 395 for communication with the restart service 380."]; and
- loading and creating an initial input/output control signal interface that establishes the message passing interface [§0035, "During OS initialization 410, the kernel mode watchdog driver 360 will load and create an initial IOCTL 390 interface with commands for establishing the message passing interface. The watchdog driver 360 will also establish an IOCTL signal interface 395 for communication with the restart service 380."].

As per dependent claim 19:

Lacombe teaches the initialization of the computer application comprises:

- linking the application with a dynamic link library [see DLL discussion §0032];
- calling the watchdog driver via the dynamic link library and through the initial input/output control signal interface to validate the message passing interface [see DLL discussion §0032]; and
- sending application location and identification information to the watchdog driver [§0015, "application information such as the relevant location and process identification is sent to the watchdog driver."].

As per dependent claim 20:

Lacombe teaches the initialization of the watchdog timer device comprises:

- setting the timer initialization value in a timer value register in the watchdog timer device [§0014, "Initialization of the watchdog driver involves loading the watchdog driver as the operating system loads following a computer system boot. During driver initialization, an initial input/output control (IOCTL) signal interface is loaded and created to establish the message passing interface."]; and
- inherently setting the counter enable bit and early warning enable bits in a control/status register in the watchdog timer device [inherently part of setting timer initialization value, see discussion §0014].

As per dependent claim 22:

Lacombe teaches the system thread must detect a periodic signal from all designated applications before initiating the command to the watchdog timer to reset the watchdog timer to the initial value [§0011, "The driver includes a system thread configured to monitor a plurality of user applications that operate in the user mode of the computer operating system. The watchdog driver also provides a first input/output control (IOCTL) signal interface for communicating control signals between the watchdog driver and one of the user applications and a second IOCTL signal interface for communicating control signals between the watchdog driver and the restart service. Lastly, a communication interface is provided for coordinating timer events with the operating system scheduler. Each timer event corresponds to one of the applications and indicates when the

application is presumed to be unresponsive"].

As per dependent claim 28:

Lacombe teaches the system watchdog and application watchdog may be selectably enabled or disabled independent of one another [see §0035, i.e., "system level watchdog time", see also §0034, see "The application watchdog driver 360" and associated discussion"].

As per dependent claim 29:

Lacombe teaches the watchdog devices are selectably configured to transmit an early warning interrupt to the CPU before the watchdog device initiates the server reset command [see §0031, "interrupt" discussion].

As per dependent claim 30:

Lacombe teaches the watchdog devices are selectably configured to transmit an early warning notification to the IOP before the watchdog device initiates the server reset command [see §0028, see e.g., "Automatic Server Recovery (ASR) watchdog found in some Compaq Computer Corporation servers" and associated discussion].

As per dependent claim 31:

Lacombe teaches the watchdog devices are selectably configured to transmit an event notification to the IOP when the watchdog device initiates the server reset command §§0011, 0012, "Each timer event corresponds to one of the applications and indicates when the application is presumed to be unresponsive.

[0012] If the system thread does not receive a message from an application within an allotted period of time, the timer event alerts the watchdog driver that the allotted time has elapsed and the watchdog driver signals the restart service to restart that application".

Claim 1 is rejected under 35 U.S.C. § 102(b) as being anticipated by **Hauck et al.** (U.S. Patent 6,026,454)

As per independent claim 1:

Hauck teaches a computer system, comprising at least one processor [microprocessor 41, col. 4, line 62], a system memory coupled to said processor [RAM 43, ROM 44, col. 5, lines 1-2], at least one input/output device coupled to said processor [e.g., see "PCMCIA modem card" col. 4, line 31], and a watchdog timer device [watchdog circuit 47, col. 5, line 11], wherein the computer system executes:

- an operating system with inherently at least two protection layers [i.e., an inherent kernel level (i.e., system level) and user level (i.e., application level – see TSR terminal server program that awaits a user response, col. 8, lines 48-50) layers or rings, col. 8, line 7];
- one or more key computer applications [extended services server program, col., 9, line 55; also see col. 12, lines 35-36]; and
- an application watchdog driver ["watchdog driver program", col. 12, line 57] that monitors user designated computer applications for periodic messages wherein if the watchdog driver receives a periodic message from all user-designated computer applications in a predetermined period of time, the watchdog driver delivers a command to clear the watchdog timer device [e.g., see "watchdog driver" and associated discussion col. 12, beginning line 37, discussion cont'd col. 13].

Indication of Allowable Subject Matter:

Dependent claim 21 appears to be allowable over the prior art of record if rewritten to include all of the limitations of the base claim and any intervening claims, subject to the results of a final search. Claim 21 stands objected to as being dependent upon a rejected base claim.

As per dependent claim 21:

The prior art of record does not teach, nor fairly suggest early warning messages that are NMI and SMI interrupts sent 9 seconds before the watchdog timer device expires, as claimed.

Claims 9-12 appear to be allowable over the prior art of record, subject to the results of a final search, for at least the following reasons:

As per independent claim 9:

The prior art of record does not teach, nor fairly suggest a watchdog driver comprising:

- a control and status register that comprises:
 - a bit for enabling the application watchdog,
 - a bit for counter reset,
 - bit fields for enabling early expiration warnings, and
 - bit fields for early expiration warning signals,
- wherein if the watchdog counter does not receive a timer reset command from the watchdog driver and the early expiration warnings are enabled, the counter is configured to transmit early expiration warnings to the rest of the computer system before the counter expires, as claimed.

Application/Control Number:
09/932,541
Art Unit: 2194

Page 17

Prior Art not relied upon

Please refer to the references listed on the attached PTO-892 which are not relied upon in the claim rejections detailed above.

Application/Control Number:
09/932,541
Art Unit: 2194

Page 18

How to Contact the Examiner:

Any inquiry concerning this communication or earlier communications from the examiner should be directed to St. John Courtenay III, whose telephone number is 571-272-3761. A voice mail service is also available at this number. The Examiner can normally be reached on Monday - Friday, 8:00 AM - 4:30 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, An Meng-AI who can be reached on 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

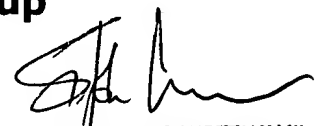
Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

All responses sent by U.S. Mail should be mailed to:

Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

**PTO CENTRAL FAX NUMBER:
703-872-9306**

- Any inquiry of a general nature or relating to the status of this application should be directed to the **TC 2100 Group receptionist: (571) 272-2100.**


**ST. JOHN COURTENAY III
PRIMARY EXAMINER**